Rozpoznanie serwera i baz

Poziom: podstawy | Dotyczy: MySQL 5.7–8.0, MariaDB 10.x–11.x | Czas: ~20 min

Przed jakąkolwiek interwencją na przejętym serwerze sprawdzamy zawsze to samo: co to za wersja, jak długo działa, ile zajmuje, kto jest podłączony i co InnoDB robi w tej chwili wewnątrz siebie.

Skróty użyte na tej stronie (1)

Skrót	Co znaczy
B2B	business to business: sprzedaż i systemy skierowane do innych firm, nie do klientów indywidualnych

Fiszka

TYLKO ODCZYT

Sprawdzenie wersji serwera: SELECT VERSION()

```
SELECT VERSION();
```

Sprawdzamy dokładny numer wersji i to, czy odpowiada MySQL czy MariaDB, po sufiksie w wyniku.

Przykład:

```
+-----------------+
| VERSION()       |
+-----------------+
| 8.0.35          |
+-----------------+
1 row in set (0.00 sec)
```

🔒 TYLKO ODCZYT

Sprawdzenie czasu działania serwera: SHOW STATUS LIKE Uptime

```
SHOW STATUS LIKE 'Uptime';
```

Sprawdzamy, ile sekund minęło od ostatniego startu procesu — zerowanie tej wartości informuje, że gdzieś doszło do restartu, świadomego albo po awarii.

Przykład:

```
+---------------+--------+
| Variable_name | Value  |
+---------------+--------+
| Uptime        | 362845 |
+---------------+--------+
1 row in set (0.00 sec)
```

🔒 **show** = FortiOS: wypisuje konfigurację, w podstawowej postaci tylko ustawienia różniące się od domyślnych, a z wariantem full-configuration także wartości domyślne

🔒 TYLKO ODCZYT

Sprawdzenie rozmiaru wszystkich baz danych: information_schema.tables

```
SELECT table_schema AS baza,
       ROUND(SUM(data_length + index_length) / 1024 / 1024, 1) AS rozmiar_mb
FROM information_schema.tables
GROUP BY table_schema
ORDER BY rozmiar_mb DESC;
```

Liczymy sumę danych i indeksów każdej bazy w megabajtach — pierwsze pytanie przy planowaniu backupu albo migracji, kiedy trzeba wiedzieć, ile realnie danych jest do przeniesienia.

Przykład:

```
+--------------------+------------+
| baza               | rozmiar_mb |
+--------------------+------------+
```

```
| produkcja          |     4821.3 |
| mysql              |        8.2 |
| performance_schema |        0.0 |
| information_schema |        0.1 |
+--------------------+------------+
4 rows in set (0.14 sec)
```

🔒 TYLKO ODCZYT

Sprawdzenie rozmiaru tabel w jednej bazie: information_schema.tables z WHERE

```
SELECT table_name AS tabela,
       table_rows AS wierszy_przyblizonych,
       ROUND(data_length / 1024 / 1024, 1) AS dane_mb,
       ROUND(index_length / 1024 / 1024, 1) AS indeksy_mb
FROM information_schema.tables
WHERE table_schema = 'NazwaBazy'
ORDER BY data_length DESC
LIMIT 10;
```

Rozbijamy rozmiar bazy na poszczególne tabele, żeby znaleźć te największe — `table_rows` dla InnoDB jest wartością szacunkową ze statystyk optymalizatora, nie dokładnym `COUNT(*)`.

Przykład:

```
+-------------+----------------------+---------+------------+
| tabela      | wierszy_przyblizonych | dane_mb | indeksy_mb |
+-------------+----------------------+---------+------------+
| zamowienia  |                48213 |   589.2 |      112.4 |
| klienci     |                 9284 |    42.1 |       18.3 |
+-------------+----------------------+---------+------------+
2 rows in set (0.09 sec)
```

🔒 TYLKO ODCZYT

Sprawdzenie liczby aktywnych połączeń: SHOW STATUS LIKE Threads_connected

```
SHOW STATUS LIKE 'Threads_connected';
```

Sprawdzamy, ile połączeń jest aktualnie otwartych, w porównaniu z limitem `max_connections` opisanym na stronie my.cnf: konfiguracja serwera.

Przykład:

```
+------------------+-------+
| Variable_name    | Value |
+------------------+-------+
| Threads_connected | 24    |
+------------------+-------+
1 row in set (0.00 sec)
```

show = FortiOS: wypisuje konfigurację, w podstawowej postaci tylko ustawienia różniące się od domyślnych, a z wariantem full-configuration także wartości domyślne

TYLKO ODCZYT

Wylistowanie wszystkich aktywnych procesów: SHOW PROCESSLIST

```
SHOW PROCESSLIST;
```

Zwracamy każde otwarte połączenie z jego bieżącym stanem i wykonywanym zapytaniem — bez uprawnienia `PROCESS` widzimy wyłącznie własne połączenia, z tym uprawnieniem widzimy wszystkie. Szersze omówienie i kasowanie konkretnych sesji rozwija strona Sesje, długie zapytania i blokady.

Przykład:

```
+----+----------+-----------+-----------+---------+------+----------
+-----------------------+
| Id | User     | Host      | db        | Command | Time | State    |
Info                  |
+----+----------+-----------+-----------+---------+------+----------
+-----------------------+
|  5 | nazwa_usera | app01:44210 | produkcja | Query   |    3 | Sending data | SELECT *
FROM zamowienia |
|  8 | nazwa_usera | localhost | NULL      | Sleep   |   42 |
|                       |
```

```
+----+----------+-----------+-----------+---------+------+----------
+-----------------------+
2 rows in set (0.00 sec)
```

show = FortiOS: wypisuje konfigurację, w podstawowej postaci tylko ustawienia różniące się od domyślnych, a z wariantem full-configuration także wartości domyślne

TYLKO ODCZYT

Podgląd wewnętrznego stanu silnika InnoDB: SHOW ENGINE INNODB STATUS

```
SHOW ENGINE INNODB STATUS\G
```

> **Pojęcie: SHOW ENGINE INNODB STATUS.** Polecenie zwracające wieloekranowy zrzut wewnętrznego stanu InnoDB w danej chwili — sekcję `TRANSACTIONS` z listą aktywnych transakcji i ich blokad, sekcję `LATEST DETECTED DEADLOCK` z ostatnim wykrytym zakleszczeniem, sekcję `BUFFER POOL AND MEMORY` z wykorzystaniem bufora i sekcję `LOG` ze stanem redo logu. To najgęstsze źródło informacji o tym, co InnoDB naprawdę robi, ale format tekstowy nie nadaje się do parsowania w skryptach — do tego lepiej sięgnąć po widoki `performance_schema` opisane na stronie Sesje, długie zapytania i blokady.

Przykład (fragment):

```
=====================================
2026-03-14 09:41:02 0x7f2a1c3fe700 INNODB MONITOR OUTPUT
=====================================
Per second averages calculated from the last 15 seconds
-----------------
BUFFER POOL AND MEMORY
-----------------
Total large memory allocated 8606711808
Buffer pool size   524288
Free buffers       12044
Database pages     511890
Modified db pages   842
Pending reads 0
```

```
Pending writes: LRU 0, flush list 0, single page 0
Pages made young 128420, not young 0
0.00 youngs/s, 0.00 non-youngs/s
Buffer pool hit rate 1000 / 1000
```

show = FortiOS: wypisuje konfigurację, w podstawowej postaci tylko ustawienia różniące się od domyślnych, a z wariantem full-configuration także wartości domyślne

TYLKO ODCZYT

Sprawdzenie ogólnych liczników pracy serwera: SHOW STATUS

```
SHOW STATUS LIKE 'Com_%';
```

Sprawdzamy liczniki wykonanych poleceń według typu (`Com_select`, `Com_insert`, `Com_update`) od ostatniego startu serwera — pierwszy krok do oceny profilu obciążenia (czy przeważa odczyt, czy zapis) bez sięgania po zewnętrzne narzędzia monitoringu.

Przykład:

```
+---------------+--------+
| Variable_name | Value  |
+---------------+--------+
| Com_select    | 984213 |
| Com_insert    |  48291 |
| Com_update    |  12043 |
| Com_delete    |    892 |
+---------------+--------+
4 rows in set (0.00 sec)
```

show = FortiOS: wypisuje konfigurację, w podstawowej postaci tylko ustawienia różniące się od domyślnych, a z wariantem full-configuration także wartości domyślne

Z życia wzięte

Z ŻYCIA WZIĘTE

Przejmujemy serwer my01 po zewnętrznym dostawcy sklepu B2B i pierwsze pytanie brzmi, ile w ogóle waży baza `sklep_b2b`. Zapytanie do `information_schema.tables` pokazuje, że jedna

> tabela z historią koszyków zajmuje więcej miejsca niż wszystkie tabele zamówień razem wzięte. Okazuje się, że nikt nigdy nie ustawił czyszczenia starych, porzuconych koszyków.

Skąd bierzemy te informacje

Rozpoznanie serwera MySQL opiera się na trzech źródłach, które wracają na tej stronie wielokrotnie. information_schema odpowiada na pytania o metadane — ile jest baz, ile ważą, jakie mają tabele — bez wpływu na sam serwer, bo to zwykłe `SELECT` do widoków katalogowych. SHOW STATUS i pokrewne polecenia `SHOW` zwracają liczniki i stan bieżący samego procesu `mysqld`, zerowane przy każdym restarcie. SHOW ENGINE INNODB STATUS schodzi jeszcze niżej, pokazując wewnętrzny stan silnika InnoDB — kolejki, transakcje, ostatni deadlock — którego żaden widok `information_schema` nie ujawnia wprost.

Częste błędy

Traktowanie table_rows jako dokładnej liczby wierszy InnoDB. Dla tabel InnoDB `information_schema.tables.table_rows` jest estymacją opartą na statystykach optymalizatora, potrafiącą różnić się od rzeczywistości o kilkanaście procent na tabelach z dużym ruchem `UPDATE`/`DELETE` — dokładną liczbę daje wyłącznie `SELECT COUNT(*)`, kosztem pełnego skanu.

Czytanie SHOW ENGINE INNODB STATUS bez \G. Bez terminatora `\G` wynik wraca jako jedna szeroka kolumna tekstu, praktycznie nieczytelna w standardowym terminalu — terminator pionowy jest tu obowiązkowy, nie kosmetyczny.

Powiązane: Architektura MySQL i MariaDB: silniki składowania, pliki, procesy · Sesje, długie zapytania i blokady · my.cnf: konfiguracja serwera

Źródła: dokumentacja MySQL 8.0, rozdział „The INFORMATION_SCHEMA TABLES Table" i „SHOW ENGINE Statement"

Revision #41
Created 2026-09-25 00:00:33 UTC by Tomasz Konieczny
Updated 2026-10-26 14:08:19 UTC by Tomasz Konieczny